QuickBLASTP: Faster Protein Alignments

Tom Madden and Greg Boratyn

Information Engineering Branch, National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health

Abstract

A sequence similarity search compares a query to a database of sequences in order to establish a relationship between the query and some of the database sequences. In many cases the search can help identify the function of a query. Basic Local Alignment Search Tool (BLAST) is a popular program used to perform similarity searches [1,2]. Here, we discuss a new implementation of the BLASTP program that compares a protein query to a protein database.

BLASTP implements a number of heuristics that make it very fast, but the popular NCBI nr database doubles in size roughly every two years. Additionally, many of the top matches found with a BLASTP search of nr are very strong, which should make them easy to identify. At the same time, high-throughput sequencing of genomes has resulted in many predicted coding regions that need to be confirmed and/or annotated by alignment to very similar sequences.

QuickBLASTP adds an initial step to identify database sequences ("candidates") that might be similar to the query. The new step makes use of an alignment free method that counts the number of 5-mers in the query sequence and quickly checks the database for sequences with a similar profile. QuickBLASTP then performs a standard BLASTP search against the candidate database sequences.

We compare QuickBLASTP to BLASTP and show that QuickBLASTP performs well in finding the strongest BLASTP matches in a fraction of the time.

Experiments

In order to compare QuickBLASTP to BLASTP, we search two sets of proteins against nr. Our goal is to demonstrate the differences in results and run times between the two programs. This experiment also demonstrates how QuickBLASTP could be used to quickly analyze a large set of proteins. First, we searched the 4,799 proteins from a WGS project (MTPK01) for *Shigella flexneri*. Second, we searched the 4,246 proteins from a transcriptomic analysis (GFAA01) of the salivary glands of the tick *Amblyomma sculptum*. We searched proteins from each project against a recent copy of the nr database with both QuickBLASTP and BLASTP. The nr database contained about 46 billion residues and 128 million sequences. Using BLASTP as the standard, we counted the matches missed by QuickBLASTP. We only examined matches with an expect value of 10^{-6} or better. Figures 1 and 2 plot the number of matches found for the two projects at different percent identity levels for both programs. Figures 3 and 4 present data on the first ten matches found for each query. The relative speedup for the searches are presented in Table 1.

The BLASTP command line was:

```
blastp -db BLASTDB/nr -query $QUERY.fsa -task blastp-fast -outfmt "7 qseqid sseqid pident length mismatch gapopen
qstart qend sstart send evalue bitscore score qlen" -parse_deflines -num_threads 8
```

The QuickBLASTP command-line was:

```
kblastp -db BLASTDB/nr -query $QUERY.fsa -candidates 1500 -thresh 0.10 -min_hits 1 -outfmt "7 qseqid sseqid pident
length mismatch gapopen qstart qend sstart send evalue bitscore score qlen" -parse_deflines  -num_threads 8
```

Figure 1

Total number of matches found by QuickBLASTP and BLASTP with a given percent identity or higher. Only matches with an expect value of 10^{-6}or better were included. The query set is the 4,799 *Shigella flexneri* proteins from the MTPK01 WGS project.

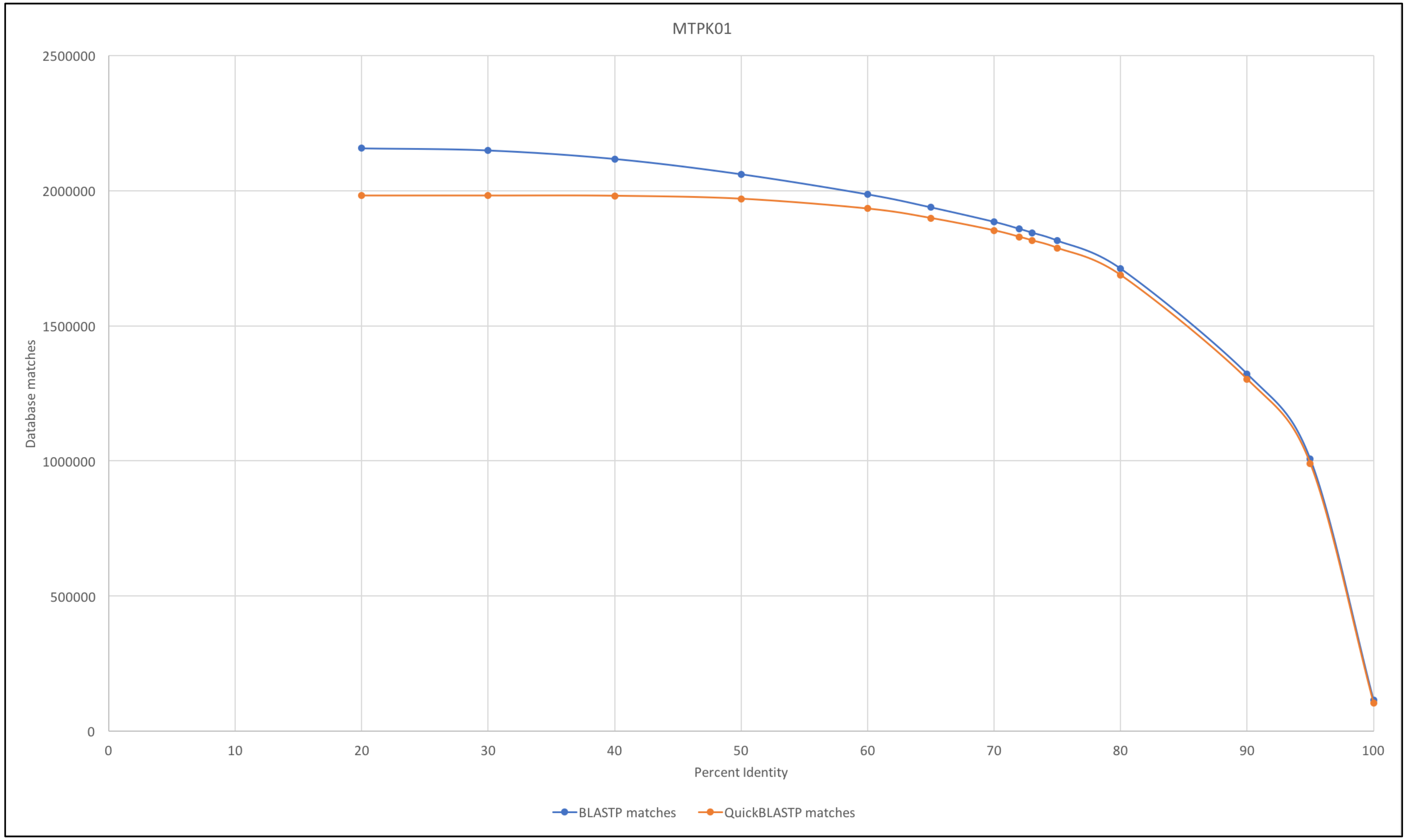


Figure 2

Total number of matches found by QuickBLASTP and BLASTP with a given percent identity or higher. Only matches with an expect value of 10^{-6}or better were included. The query set is the 4,246 proteins from the GFAA01 transcriptomic analysis of the salivary glands of the tick *Amblyomma sculptum*.

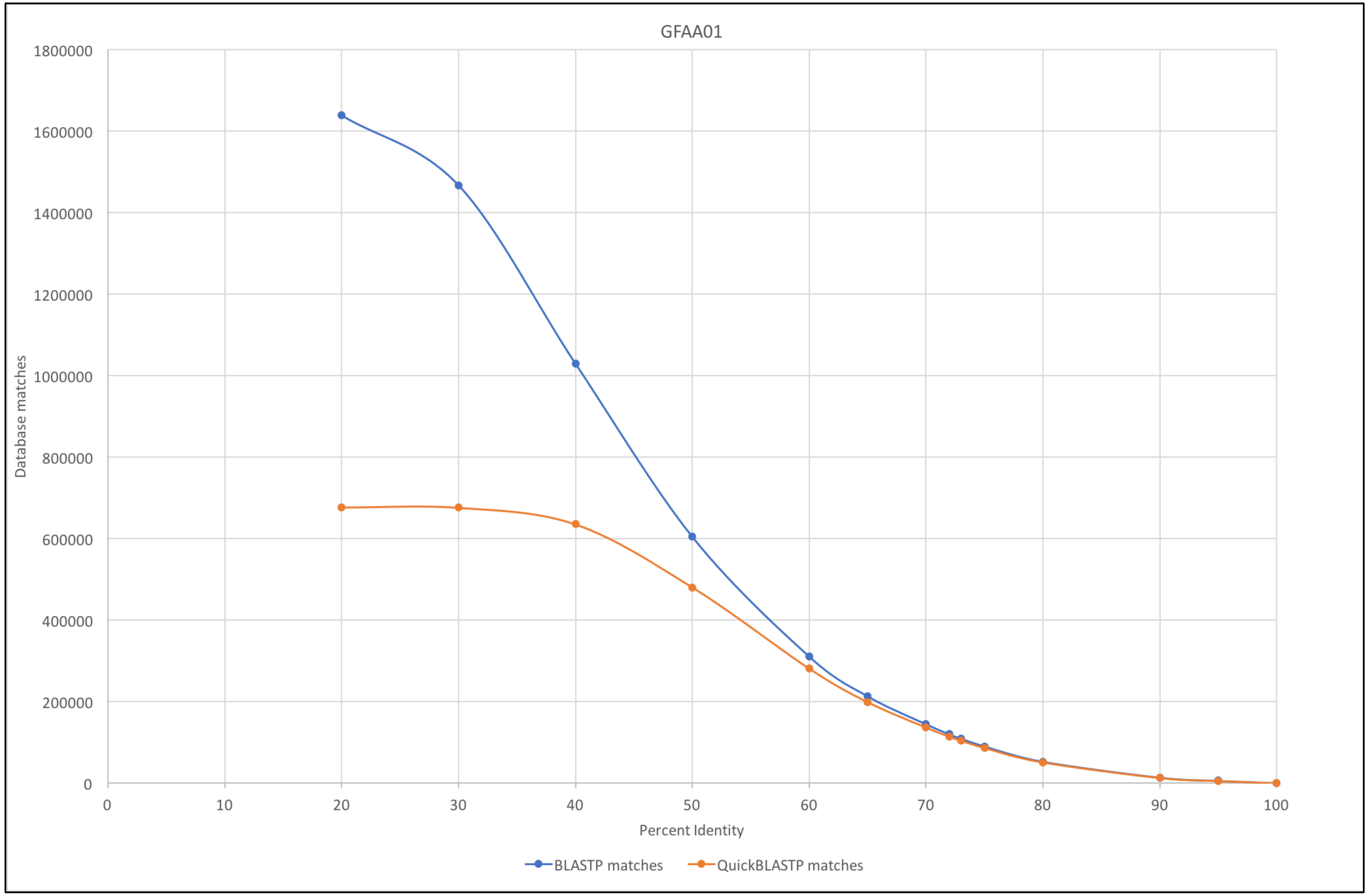


Figure 3

Missed matches in a QuickBLASTP search of MTPK01. MTPK01 proteins were searched with QuickBLASTP and BLASTP against nr and the top 10 matches per query examined. The blue bars present the number of matches with an expect value of 10^{-6} or better that BLASTP found but QuickBLASTP did not. The orange bars present the same data but only consider matches with 65% or more identity between the query and subject sequences.

The MTPK01 WGS project sequenced *Shigella flexneri*. The MTPK01 project contains 4,799 proteins.

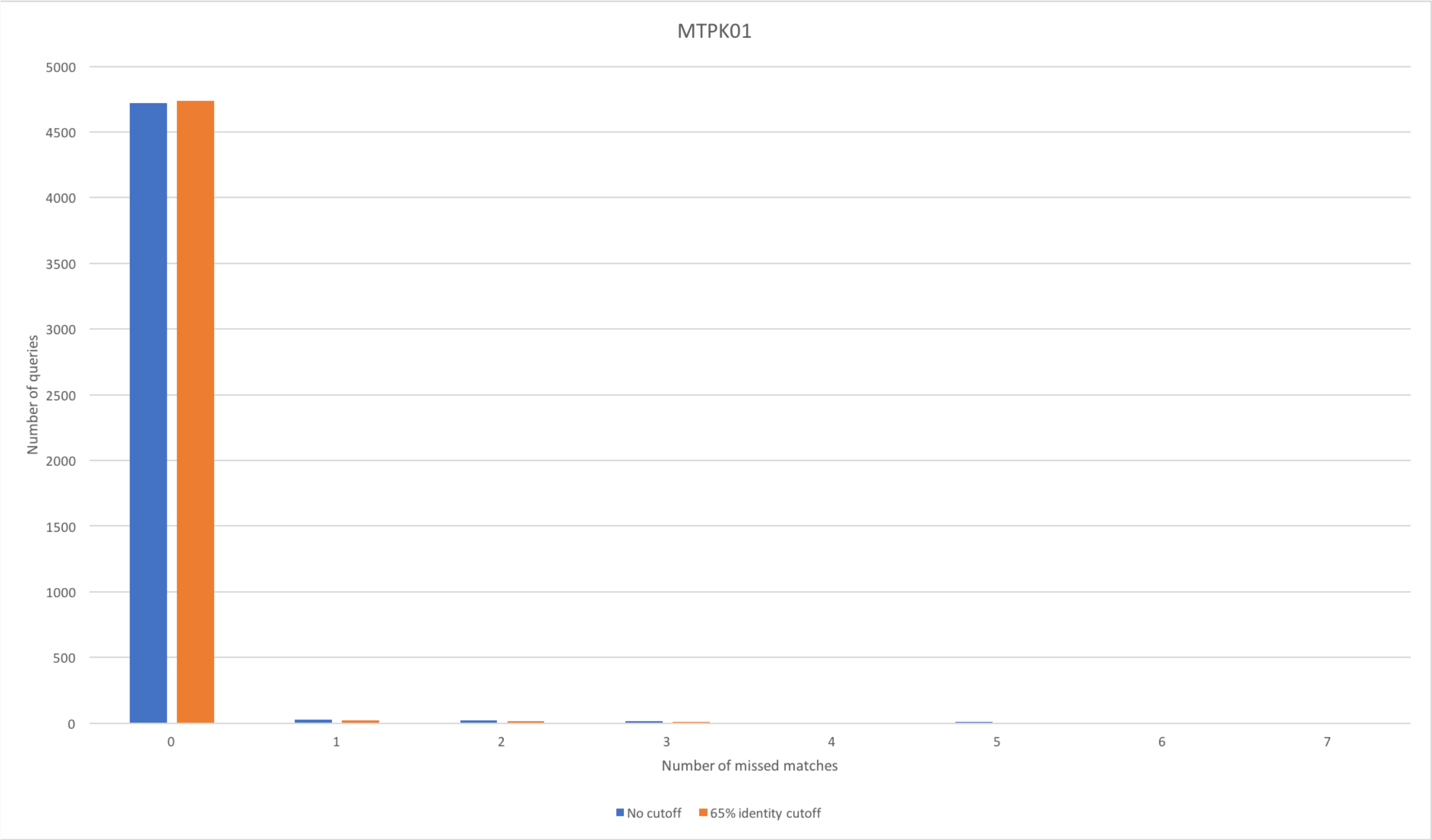


Figure 4

Missed matches in a QuickBLASTP search of GFAA01. GFAA01 proteins were searched with QuickBLASTP and BLASTP against nr and the top 10 matches per query examined. The blue bars present the number of matches with an expect value of 10^{-6} or better that BLASTP found but QuickBLASTP did not. The orange bars present the same data but only consider matches with 65% or more identity between the query and subject sequences.

GFAA01 is a transcriptomic analysis of the salivary glands of the tick *Amblyomma sculptum*. The GFAA01 project contains 4,246 proteins.

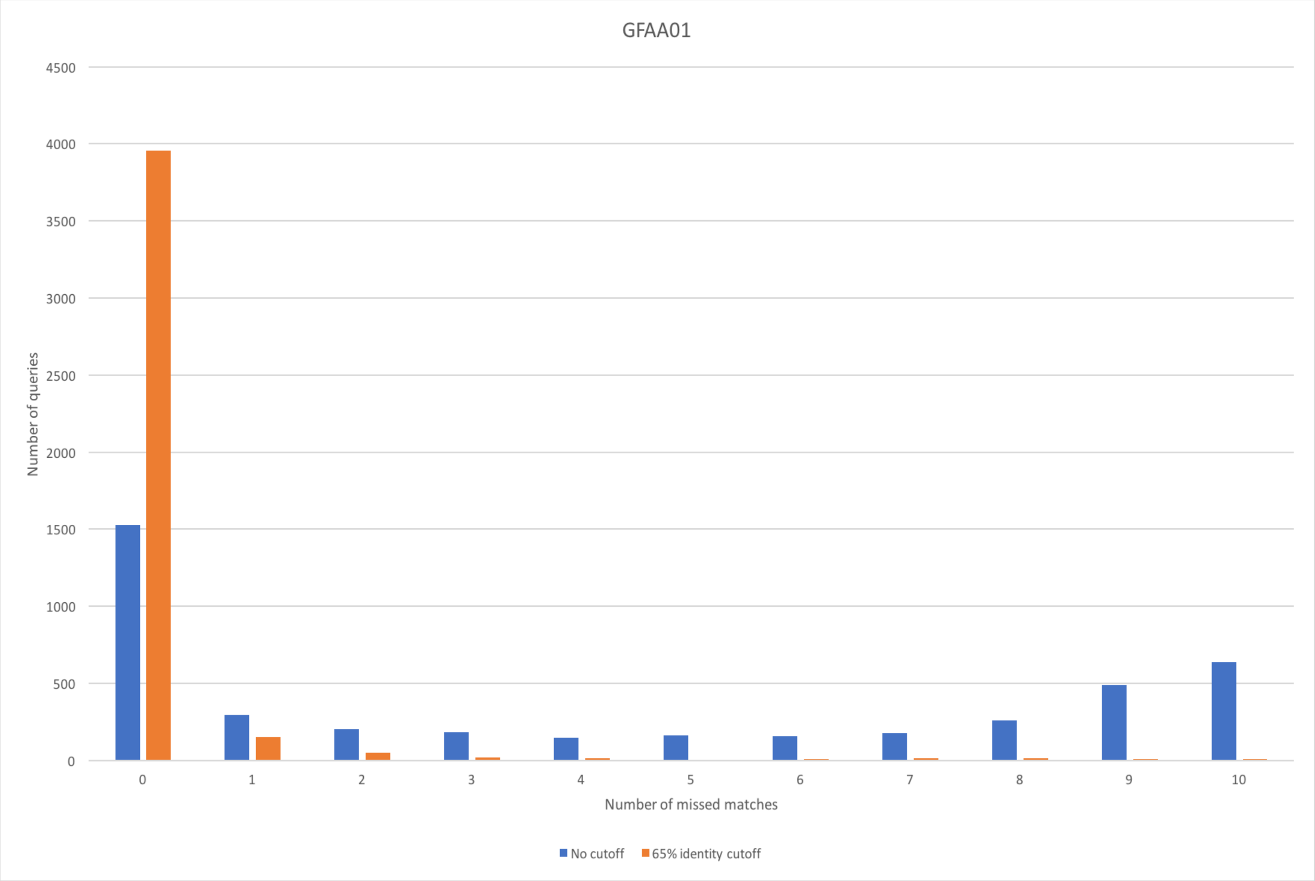


Table 1

Wall-clock times to search proteins from two projects with BLASTP and QuickBLASTP. Searches were run with eight CPUs using a recent copy of the nr database. Multiple runs were performed and the run with the lowest time was selected. See text for search parameters.

Project	BLASTP (seconds)	QuickBLASTP (seconds)	Speedup	Number of queries
MTPK01	36,527	2,906	12.6	4,799
GFAA01	64,575	3,185	20.3	4,246

Methods

Goal: Quickly identify similar proteins with an alignment free method.

Procedure: Fingerprint a protein by sampling k-mers.

Criteria: The Jaccard similarity coefficient [3]:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

$$0 \leq J(A,B) \leq 1$$

The Jaccard similarity coefficient is the intersection of two sets over the union of those two sets. Our process is based on the one outlined in [4].

Building the index for a BLAST database

For every protein in the database to be indexed:

1. Translate the residues into a reduced alphabet (figure 5)
2. Break the protein into 150 residues chunks that overlap the neighboring chunk by 50 residues.
3. Calculate a hash value on all 5-mers in a chunk and save the lowest 32 values as a data array.
4. Create an additional index to speed up retrieval.

Searching the index with a protein query.

1. Translate the query into a reduced alphabet (figure 5).
2. Break up query into 150 residue chunks (with overlap) and apply the same hashing procedure as above to produce query data arrays.
3. Identify database chunks similar to query chunks.
4. Estimate the Jaccard index between a query chunk and the database chunk data arrays by counting the number of identical hash values (figure 6).
5. Perform a BLASTP search between the query and the top 1500 subject sequences that have a Jaccard index of 0.1 or more.

Figure 5

Reduced protein alphabet used for alignment free calculations. The reduced alphabet replaces the 23 residues with 15 groups. Use of the reduced alphabet improves the sensitivity of the procedure. This reduced alphabet was suggested in [5].

ST

IJV

LM

KR

EQZ

BD

P

G

F

Y

C

W

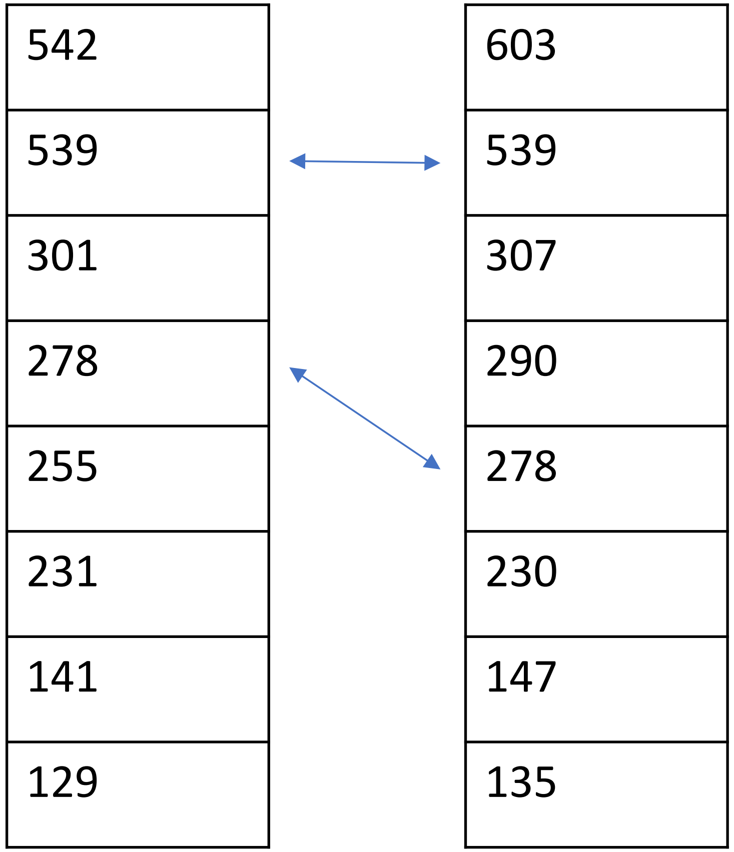
A

H

N

Figure 6

Estimation of the Jaccard index with data arrays. This figure presents the procedure with two eight element arrays. Each element of the array contains an integer that is the result of computing a hash value on a 5-mer from a protein sequence. The program scans the arrays to find common hash values. In this example, there are two matches out of eight, so the Jaccard index is estimated as 0.25. Normally, a data array contains 32 elements.



542	603
539	539
301	307
278	290
255	278
231	230
141	147
129	135

Availability

- QuickBLASTP can be run by selecting the Protein BLAST link at the NCBI BLAST page (blast.ncbi.nlm.nih.gov) and using the Quick BLASTP algorithm.
- SmartBLAST also uses the QuickBLASTP algorithm to quickly search the nr database.

Acknowledgements

We’d like to thank George Colouris for originally suggesting the current approach in the context of neighboring proteins. We’d also like to thank Richa Agarwala for suggestions that improved the approach. Finally, we’d like to thank David Lipman for originally suggesting the project and encouraging our work.

This research was supported by the Intramural Research Program of the NIH, National Library of Medicine.

References

1.) Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 1997 Sep 1;25(3389-402)

2.) Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, Madden TL. BLAST+: architecture and applications. BMC Bioinformatics. 2009 Dec 15;10:421.

3.) https://en.wikipedia.org/wiki/Jaccard_index

4.) http://infolab.stanford.edu/~ullman/mmds/ch3.pdf

5.) Shiryev SA, Papadopoulos JS, Schäffer AA, Agarwala R. Improved BLAST searches using longer words for protein seeding. Bioinformatics 2007 Nov 1;23(21):2949-51.